Spotipy – Analyzing the Top Hits

Project Members: Cam Foster, Anis Ali, Michael Alread, Ida Astaneh

Release Date: 2020-10-27

# The Original Data

<Need to add stuff here>

# Process Overview

To expand our original dataset and get the audio features for each track we needed clean the dataset to filter for only songs in the US region for 2017 extract just the unique track IDs. That allowed us to call the API fewer times and reduce the time it took to get our data.

Once we had the audio features for all songs in our final dataset, we merged the original dataset with the audio features dataset on track ID giving us our final, working dataset.

<Talk about the analysis process>

# Cleaning the Data

# Spotify API

There is a python library called spotipy that is used to interact with the Spotify Web API. You are required to create a developer account and get a Client ID and Client Secret key to authenticate with Spotify. Most of the functions in this library are geared towards interacting with Spotify for web applications and doing things such as selecting the next track, getting related artists, and get current user data. We used two functions, spotipy.track() and spotipy.audio\_features(), to get information about the track requested.

A function was written to call the API and return the track identifiers (i.e. artist, track name, album name, etc.) and audio features (i.e. loudness, acousticness, energy, etc.). We ran into an issue with the API timing out but adding a pause under an except clause gave us a work around for this issue.

All of the audio features are defined by Spotify on their developer website.

# Questions and Hypothesis

## Given the data provided, is there a direct correlation between our Spotipy headers?

## Could we bucket the stream counts to show the percentage breakdown? How “popular” is popular?

## Which artist/album appears the most and least amount of times on the list?

## What decade, prior to 2010, was responsible for providing the most streams in 2017?